

ОГАОУ ДПО «Белгородский институт развития»

Рассмотрено  
цикловой комиссией  
«Автоматика и мехатронные комплексы»  
Протокол заседания № 12  
от «28» января 2019 г.  
Председатель цикловой комиссии  
\_\_\_\_\_ Недоступенко Д.А.

Учебно-методическое пособие  
для выполнения лабораторных работ по  
ПМ 02. «Техническое обслуживание, ремонт и испытание мехатронных  
систем»  
МДК 02.01 «Техническое обслуживание, ремонт и испытание  
мехатронных систем»

специальность  
15.02.10 «Мехатроника и мольная робототехника»

Недоступенко Дарья Александровна,  
преподаватель ОГАОУ  
«Белгородский индустриальный колледж»

Белгород  
2019

## Содержание

Пояснительная записка	3
Приложение 1. Лабораторная работа №1 Управление led-светодиодом	5
Приложение 2. Лабораторная работа №2 Управление rgb led светодиодом	7
Приложение 3. Лабораторная работа №3 Изучение блоков управления	9
Приложение 4. Лабораторная работа №4 Программирование «перо и звук»	12
Приложение 5. Лабораторная работа №5 Изучение датчиков	14
Приложение 6. Лабораторная работа №6 Типы данных. Логические операторы. Оператор If-else, for-next, while, do-while. Мигание led светодиодом	17
Приложение 7. Лабораторная работа №7а Управление rgb led светодиодом. Управляющие операторы switch-case	20
Приложение 8. Лабораторная работа №7б Arduino ide: программирование серводвигателя	23
Приложение 9. Лабораторная работа №8 Операции по перемещению манипулятора. Задача прямой кинематики	25
Приложение 10. Лабораторная работа №9 Операции по перемещению манипулятора. Задача обратной кинематики	27
Приложение 11. Лабораторная работа №10а Простые операции по перемещению объектов в пространстве с помощью манипулятора	29
Приложение 12. Лабораторная работа №10б Arduino ide: простые операции по перемещению объектов в пространстве с помощью манипулятора	31
Приложение 13. Лабораторная работа №11-12 Изучение принципов работы rgb датчика. Распознавания цвета. Программирование оптического датчика. Применение оптического датчика в алгоритмах управления роботом - манипулятором	34
Приложение 14. Лабораторная работа №13 Взаимодействие робота и датчиков. Программирование сортировочной установки	36
Приложение 15. Лабораторная работа №14 Программирование датчика влажности. Применение датчика влажности в алгоритмах управления роботом - манипулятором»	38
Приложение 16. Лабораторная работа №15 Программирование потенциометра. Применение потенциометра в алгоритмах управления роботом - манипулятором	40
Приложение 17. Правила техники безопасности при работе обучающихся в лабораториях	42
Приложение 18. Критерии оценки выполнения лабораторной работы	44
Библиографический список	45

## Пояснительная записка

Учебно-методическое пособие по выполнению лабораторных работ по профессиональному модулю ПМ. 02 «Техническое обслуживание, ремонт и испытание мехатронных систем» МДК 02.01 «Техническое обслуживание, ремонт и испытание мехатронных систем» предназначено для обучающихся 2 курса специальности 15.02.10 «Мехатроника и мобильная робототехника». Пособие составлено с учетом требований Федерального государственного образовательного стандарта среднего профессионального образования (далее ФГОС СПО) и необходимостью учебно-методического обеспечения междисциплинарного курса МДК 02.01 «Техническое обслуживание, ремонт и испытание мехатронных систем».

Актуальность определяется требованиями ФГОС. Основная задача пособия сводится к ознакомлению с базовыми принципами работы контроллеров и манипуляторов в целом. Кроме того, работа с реальным оборудованием вызывает интерес и стремление у обучающихся к изучению специальных дисциплин.

Целью представленных материалов является формирование общих (ОК 01-05, 10) и профессиональных (ПК 2.1-2.3) компетенций обучающихся при выполнении лабораторных работ.

Задачи:

- активизация мыслительной деятельности и познавательного интереса обучающихся;
- формирование умений применять полученные теоретические знания на практике, реализация единства теоретической и практической деятельности;
- выработка при решении поставленных задач профессионально - значимых качеств: самостоятельность, точность, творческая инициатива.

Изучение междисциплинарного курса МДК 02.01 «Техническое обслуживание, ремонт и испытание мехатронных систем» базируется на знаниях и умениях, приобретенными обучающимися при изучении курсов электротехники, информатики, основ автоматического управления и математики.

Основным объектом изучения в рамках выполнения лабораторных работ является робот-манипулятор Оптима-1 с микроконтроллером Arduino UNO. Отработка навыков программирования на общедоступной плате с широкой поддержкой по всему миру поможет унифицировать полученные знания. Применение алгоритмов на конкретном роботе позволит более наглядно понять некоторые принципы работы автономных устройств. Среда разработки Arduino IDE предоставляет язык программирования, схожий с C++, что позволит обучающемуся без особого труда перейти на другие языки с похожим синтаксисом.

Представленное учебно-методическое пособие для выполнения лабораторных работ состоит из трех блоков.

Первый блок, включает 15 лабораторных работ по изучению

роботизированных систем на базе робота – манипулятора «ОРТИМА».

Лабораторные работы, приведенные в учебно-методическом пособии, включают в себя следующие элементы: наименование темы, цель занятия, формируемые у обучающихся знания и умения, перечень формируемых общих и профессиональных компетенции, время выполнения работы, ход работы, порядок выполнения работы, краткие теоретические сведения, практическую часть (указания по выполнению), контрольные вопросы.

В теоретической части приводится структурированный материал, необходимый для подготовки обучающихся к лабораторным работам. Практическая часть представляет собой проведение опытов и экспериментов. В лабораторных работах отражены задания повышенного уровня, отмеченные знаком \*. Вопросы к каждой лабораторной работе позволяют провести контроль обучающихся. На контрольные вопросы обучающиеся отвечают письменно в технических отчетах по лабораторным работам. Для проведения следующей лабораторной работы требуется сдача с устной защитой технического отчета по предыдущей работе.

Второй блок, включает в себя правила техники безопасности при работе обучающихся в лабораториях. Обучающиеся допускаются преподавателем к выполнению лабораторной работы только после инструктажа по технике безопасности работы.

Второй блок носит теоретический характер и содержит критерии оценки выполнения лабораторной работы.

Учебно – методическое пособие для выполнения лабораторных работ по ПМ 02. «Техническое обслуживание, ремонт и испытание мехатронных систем» МДК 02.01 «Техническое обслуживание, ремонт и испытание мехатронных систем» рекомендовано преподавателям технических специальностей по программам подготовки специалистов среднего звена по специальностям 27.02.04 Автоматические системы управления и 15.02.10 Мехатроника и мобильная робототехника.

Таким образом, учебно – методическое пособие позволяет реализовать формирование общих и профессиональных компетенций обучающихся при выполнении лабораторных работ по профессиональному модулю ПМ.02 «Техническое обслуживание, ремонт и испытание мехатронных систем».

## ЛАБОРАТОРНАЯ РАБОТА №1 «УПРАВЛЕНИЕ LED-СВЕТОДИОДОМ»

**Цель работы:** познакомиться с принципами управления устройствами вывода на примере светодиода.

**При выполнении лабораторной работы обучающийся должен:**

**Знать:** последовательность подключения светодиода и резистора в схеме; границы восприятия светодиодного освещения; правила управления светодиодами.

**Уметь:** читать электросхемы; осуществлять технический контроль качества технического обслуживания; анализировать задачу и/или проблему и выделять её составные части.

**Формируемые компетенции:**

ОК 01 Выбирать способы решения задач профессиональной деятельности, применительно к различным контекстам;

ОК 03 Планировать и реализовывать собственное профессиональное и личностное развитие;

ПК 2.1 Осуществлять техническое обслуживание компонентов и модулей мехатронных систем и мобильных робототехнических комплексов в соответствии с технической документацией.

**Время выполнения работы:** 1 час 30 минут.

**Ход работы:**

1. Проведение преподавателем инструктажа по технике безопасности;
2. Изучить краткие теоретические сведения и законспектировать в технический отчет;
3. Последовательно выполнить порядок выполнения работы;
4. Письменно ответить на контрольные вопросы.

### Краткие теоретические сведения

Светодиод (или LED) – источник света, постепенно замещающий в наши дни старые лампы накаливания. Они потребляют меньше энергии и позволяют получить более «чистый» свет. Из-за высокой яркости одного светодиода его можно сделать очень маленьких размеров. Так, например, на экранах современных телевизоров и мониторов могут одновременно работать несколько десятков миллионов светодиодов. С точки зрения обывателя всё же есть некоторая разница

В использовании традиционных ламп накаливания и LED. Светодиод должен быть подключен одним концом к катоду (минус), а другим к аноду (плюс), и никак не наоборот. Чтобы зажечь LED необходимо всего лишь подать ток.

На платах Arduino есть встроенный светодиод, и чаще всего он подключен к контакту (13). Второй контакт при этом должен быть соединён с «землёй» (GND). Осталось только понять, как написать программу, которая будет включать светодиод.

Для начала познакомимся с некоторыми правилами написания программ для микроконтроллеров Arduino. Любая программа должна содержать две конструкции (функции): *setup()* и *loop()*. Первая часть будет выполняться при запуске программы (*setup* – установка, настройка), а вторая повторяться циклично (*loop* – петля). Как правило, в функции *setup()* содержится информация о том, какие контакты для чего используются. В нашем случае необходимо указать, что (13) контакт мы будем использовать только для вывода *pinmode(13, OUTPUT)*;

Теперь можно подать ток на контакт (13). Для этого воспользуемся командой *digitalWrite(13, HIGH)*; Для отключения LED необходимо вместо HIGH написать LOW. Одной из самых используемых команд на практике является команда ожидания *delay()*; В скобках указывается длительность паузы в миллисекундах.

```
void setup(){
  pinMode(13, OUTPUT);
}
void loop(){
  digitalWrite(13,HIGH);
  delay(1000);
}
```

### Порядок выполнения работы

1. Подключите робот-манипулятор к компьютеру при помощи USB-кабеля.
2. Запустите Arduino IDE. Убедитесь, что устройство обнаружено.
3. Выставьте модель платы как Arduino Uno.
4. Напишите программу по образцу. Нажмите на кнопку загрузки на плату.
5. Убедитесь, что светодиод на плате светится.
6. Напишите программу для мигания диода с частотой раз в 2 секунды (секунду горит, секунду нет).
7. \*Измените программу, чтобы светодиод горел пятую часть всего времени при той же частоте. Расчёты запишите в тетрадь.

### Контрольные вопросы

1. Почему светодиодные ленты имеют маленькие чипы на всём протяжении?
2. Светодиод может испускать разный по цвету свет в зависимости от используемого вещества. Можно ли поменять яркость света? Если да, то как?
3. *DigitalWrite()* переводится как «цифровая запись». Что объединяет тип записываемого значениями с цифровым подходом?
4. Представьте, что к контакту (12) подключен другой светодиод. Как будет выглядеть программа, включающая порознь два светодиода?

## ЛАБОРАТОРНАЯ РАБОТА №2 «УПРАВЛЕНИЕ RGB LED СВЕТОДИОДОМ»

**Цель работы:** познакомиться со световой схемой RGB и принципом работы соответствующего светодиода.

**При выполнении лабораторной работы обучающийся должен:**

**Знать:** последовательность подключения светодиода и резистора в схеме; классификацию и виды отказов оборудования; технологическую последовательность разборки, ремонта и сборки узлов и механизмов мехатронных систем.

**Уметь:** читать электросхемы; кратко обосновывать и объяснить свои действия (текущие и планируемые); составлять план действия.

**Формируемые компетенции:**

ОК 01 Выбирать способы решения задач профессиональной деятельности, применительно к различным контекстам;

ОК 04 Работать в коллективе и команде, эффективно взаимодействовать с коллегами, руководством, клиентами;

ПК 2.2 Диагностировать неисправности мехатронных систем и мобильных робототехнических комплексов с использованием алгоритмов поиска и устранения неисправностей.

**Время выполнения работы:** 1 час 30 минут.

**Ход работы:**

- 1.Проведение преподавателем инструктажа по технике безопасности;
- 2.Изучить краткие теоретические сведения и законспектировать в технический отчет;
- 3.Последовательно выполнить порядок выполнения работы;
- 4.Письменно ответить на контрольные вопросы.

### Краткие теоретические сведения

Человеческий глаз способен воспринимать свет в определённом диапазоне от красного до фиолетового. Каждому цвету радуги при этом соответствует определённая длина волны. Привычные нам цвета, не представленные в спектре радуги являются комбинацией нескольких цветов радуги, зачастую взятых в разных пропорциях. Художники издревле используют палитру для смешения имеющихся красок и получения уникального нужного цвета.

Цифровая природа вычислительных устройств не позволяет в полной мере передать всё разнообразие цветов реального мира. Существующие системы представления цветов позволяют получить разнообразные оттенки, и для большинства людей такой точности достаточно. Самой распространённой цифровой схемой представления цвета на сегодняшний день является RGB (redgreenblue – красный зелёный синий). Полученный цвет является суммой красного, зелёного и синего цветов с разной интенсивностью. Эта световая схема обязана своей популярностью устройству большинства экранов. Каждая точка (пиксель) современного

экрана содержит три маленькие лампочки (например, светодиода) красного, зелёного и синего цветов. Регулируя интенсивность каждого цвета можно получить всю палитру. Яркость каждого цвета изменяется от 0 до максимального значения в 255.

Управление RGB светодиода требует четырёх контактов. Один из них должен быть подключен к (GND), а остальные три будут управлять соответствующими каналами цвета. Модифицируем программу из предыдущей работы.

```
void setup(){
  pinMode(11, OUTPUT);
  pinMode(12, OUTPUT);
  pinMode(13, OUTPUT);
}
void loop(){
  digitalWrite(11,HIGH);
  digitalWrite(12,HIGH);
  digitalWrite(13,HIGH);
  delay(1000);
}
```

В данном случае мы включили все три светодиода. Для получения других цветов следует изменить комбинацию включенных диодов.

### Порядок выполнения работы

1. Заполните таблицу цветов 1 для разных состояний RGB LED

Таблица 1

Номер	Комбинация светодиодов	Цвет
1	Красный	Красный
2	Зеленый	Зеленый
3		
4		
5		
6		
7	Красный+зеленый+синий	Белый

2. Напишите программу для последовательного переключения между цветами таблицы.

3. Напишите программу по быстрому переключению цветов. Уменьшайте время до тех пор, пока вы не перестанете различать мерцание цветов.

### Контрольные вопросы

1. Можно ли к одному контакту платы Arduino подключить несколько светодиодов? Если да, то, когда это целесообразно?

2. \*Светодиоды подключают к плате только с дополнительным последовательным сопротивлением (Зачастую оно находится на плате



светодиода). Почему? Используют ли дополнительное сопротивление при подключении ламп накаливания?

3. В новогодних гирляндах используются цепи из светодиодов. Придумайте свой собственный режим работы для гирлянды с тремя рядами разноцветных светодиодов.

4. Человеческий глаз не может различить очень быстрое мерцание цветов. Как это можно использовать для изменения яркости LED?

### ЛАБОРАТОРНАЯ РАБОТА №3 «ИЗУЧЕНИЕ БЛОКОВ УПРАВЛЕНИЯ»

**Цель работы:** запрограммировать блок управления робота Оптима-1.

**При выполнении лабораторной работы обучающийся должен:**

**Знать:** технологическую последовательность разборки, ремонта и сборки узлов и механизмов мехатронных систем; физические принципы работы, конструкцию, технические характеристики, области применения, правила эксплуатации оборудования мехатронных систем.

**Уметь:** оформлять результаты поиска; определять актуальность нормативно-правовой документации в профессиональной деятельности; выстраивать траектории профессионального и личностного развития.

**Формируемые компетенции:**

ОК 02 Осуществлять поиск, анализ и интерпретацию информации, необходимой для выполнения задач профессиональной деятельности;

ОК 05 Осуществлять устную и письменную коммуникацию на государственном языке с учетом особенностей социального и культурного контекста;

ПК 2.3 Производить замену и ремонт компонентов и модулей мехатронных систем и мобильных робототехнических комплексов в соответствии с технической документацией.

**Время выполнения работы:** 1 час 30 минут.

**Ход работы:**

1. Проведение преподавателем инструктажа по технике безопасности;
2. Изучить краткие теоретические сведения и законспектировать в технический отчет;
3. Последовательно выполнить порядок выполнения работы;
4. Письменно ответить на контрольные вопросы.

#### **Краткие теоретические сведения**

Блоком управления называется система контроллеров и двигателей, приводящая в движение робота. Блоки управления есть у каждого автомобиля, подъемного крана и пылесоса. Задача блока управления – обеспечить согласованную работу нескольких двигателей и устройств вывода. Например, блок управления «климат-контроль» в автомобиле не будет одновременно использовать кондиционер и обогрев.

Блок управления робота-манипулятора Оптима-1 должен учитывать текущее положение робота и оценивать возможность совершения того или иного движения. Программирование контроллеров – очень сложный и трудоёмкий процесс. Многие проверочные и оценочные функции должны выполняться одновременно и одновременно оказывать влияние на работу робота.

Робот-манипулятор Оптима-1 содержит дополнительный модуль для работы с двигателями. Он позволяет одновременно управлять несколькими из них. Для управления отдельным двигателем необходимо воспользоваться

функцией *setPWM()*. В скобках указывается номер двигателя (нумерация с нуля), начальное время выполнения (за редким исключением ноль), значение двигателя, соответствующее углу поворота. Подробнее о диапазоне этих значений речь пойдёт в одной из следующих лабораторных работ.

Плата PWM в роботе-манипуляторе находится отдельно от микроконтроллера Arduino UNO и должна описываться как отдельный объект с именем *pwm*. В самом начале документа необходимо подключить соответствующие библиотеки.

```
#include<Adafruit_PWMServoDriver.h>
#include <Adafruit_TCS34725.h>
Adafruit_PWMServoDriverpwm = Adafruit_PWMServoDriver(); void
setup(){
  pwm.begin();
}
void loop(){
  pwm.setPWM(0,0,160);
  delay(1000);
  pwm.setPWM(0,0,160);
  delay(1000);
}
```

### **Порядок выполнения работы**

1. Загрузите данную программу на контроллер.
2. Опишите поведение робота. Все ли роботы выполняют одинаковые действия?
3. Составьте программу, которая будет управлять последовательно всеми доступными приводами робота.
4. Напишите программу, которая позволит схвату перейти в самую верхнюю достижимую точку. Запишите значения для каждого двигателя.
5. Добавьте блок, включающий встроенный светодиод в конце выполнения.

### **Контрольные вопросы**

1. Как в сложных системах происходит согласование блоков управления?
2. Современные устройства содержат преимущественно электронные блоки управления. Приведите примеры физических блоков управления.
3. Какие проверки необходимо делать в ходе работы блока управления робота Оптима-1? К каким последствиям может привести неправильная работа программы или некорректный дизайн блока управления?
4. В чём принципиальное отличие активации светодиода и поворота мотора с физической точки зрения?
5. Можно ли подсоединить светодиод и двигатель к одному и тому же контакту платы?

## ЛАБОРАТОРНАЯ РАБОТА №4 «ПРОГРАММИРОВАНИЕ «ПЕРО И ЗВУК»

**Цель работы:** получить допустимые значения для поворота каждого двигателя робота-манипулятора Оптима-1.

**При выполнении лабораторной работы обучающийся должен:**

**Знать:** понятие, цель и функции технической диагностики; методы диагностирования, неразрушающие методы контроля; понятие, цель и виды технического обслуживания.

**Уметь:** обнаруживать неисправную работу оборудования и принимать меры для устранения и предупреждения отказов и аварий мехатронных систем; выполнять работы по устранению недостатков, выявленных в процессе эксплуатации оборудования.

**Формируемые компетенции:**

ОК 01 Выбирать способы решения задач профессиональной деятельности, применительно к различным контекстам;

ОК 04 Работать в коллективе и команде, эффективно взаимодействовать с коллегами, руководством, клиентами;

ПК 2.2 Диагностировать неисправности мехатронных систем и мобильных робототехнических комплексов с использованием алгоритмов поиска и устранения неисправностей.

**Время выполнения работы:** 1 час 30 минут.

**Ход работы:**

1. Проведение преподавателем инструктажа по технике безопасности;
2. Изучить краткие теоретические сведения и законспектировать в технический отчет;
3. Последовательно выполнить порядок выполнения работы;
4. Письменно ответить на контрольные вопросы.

### Краткие теоретические сведения

Методом проб и ошибок можно определить необходимые для достижения заданной точки значения поворота каждого двигателя, подключенного к плате PWM. Для более плавного и точного перемещения, однако, этого недостаточно. Если для каждого двигателя будет известно значение, необходимое для поворота на известные крайние значения, то можно будет рассчитать значение для любого интересующего нас угла.

Поворот против часовой стрелки будем считать положительным. Максимальное и минимальное значение угла каждого двигателя составляет  $\pm 90$  градусов. Если мы знаем, что для поворота на  $-90$  градусов необходимо передать значение  $A$ , а для поворота на  $+90$  градусов  $B$ , то для расчёта значения для угла  $\alpha$  достаточно воспользоваться формулой

$$X=A+(B-A)*(\alpha+90)/180$$

Выяснение уникальных значений постоянной ошибки или погрешности для каждого отдельного робота или датчика называется **калибровкой**. Итак, для калибровки робота нам понадобится находить для

каждого двигателя значения поворота.

Для хранения нескольких значений в программировании используются массивы. Создаётся массив целых чисел в Arduino IDE следующим образом: `inta[5]`. Сперва пишется тип данных в массиве, затем его имя и размер в квадратных скобках. В нашем случае нам понадобятся сразу три массива целых чисел. Для получения или задания отдельного числа в массиве используется имя массива и индекс интересующего элемента в квадратных скобках. Например, следующая команда записывает в первую ячейку массива число 5. `a[1] = 5`; Также стоит отметить, что нумерация элементов идёт с нуля.

### Порядок выполнения работы

1. Составьте калибровочную таблицу 1 для робота Оптима-1

Таблица 1

Номер оси	Значение при угле 90 градусов (разжат, спрятан)	Значение при угле 90 градусов (зажат, выставлен)	Начальное значение при нулевом повороте
0			
1			
2			
3			
Схват			
Раздатчик (ключка)			

2. Напишите программу для достижения самой верхней точки схватом (манипулятор должен быть направлен вертикально вверх).

3. Изменяя в исходной программе значения для одного двигателя, выясните, при каких числах поворот в каждой оси составляет прямой угол.

4. Внесите полученные значения в массивы `minAngles`, `maxAngles`, `meanAngles`. (angle–угол)

5. Скорректируйте программу из предыдущей лабораторной работы так, чтобы один и тот же код функций `setup()` и `loop()` исполнялись одинаково на всех роботах. Калибровочные массивы при этом могут отличаться.

### Контрольные вопросы

1. Почему роботы отличаются между собой? Что можно сделать, чтобы свести разницу между ними к минимуму?

2. Достаточно ли в каждой оси возможность поворота на 180 градусов? Если нет, то приведите примеры точек, которых можно достичь с большими лимитами.

3. Как Вы понимаете концепцию универсальности программного кода?

4. Существуют ли примеры значений, при которых схват манипулятора будет расположен ниже уровня стола.

5. Как при помощи известных Вам команд можно сделать движение робота более плавным?

## ЛАБОРАТОРНАЯ РАБОТА №5 «ИЗУЧЕНИЕ ДАТЧИКОВ»

**Цель работы:** запрограммировать работу робота в зависимости от текущего показания оптического датчика.

**При выполнении лабораторной работы обучающийся должен:**

**Знать:** физические принципы работы, конструкцию, технические характеристики, области применения, правила эксплуатации оборудования мехатронных систем; порядок проведения стандартных и сертифицированных испытаний.

**Уметь:** выстраивать траектории профессионального и личностного развития; организовывать работу коллектива и команды.

**Формируемые компетенции:**

ОК 04 Работать в коллективе и команде, эффективно взаимодействовать с коллегами, руководством, клиентами;

ОК 10 Пользоваться профессиональной документацией на государственном и иностранном языке;

ПК 2.2 Диагностировать неисправности мехатронных систем и мобильных робототехнических комплексов с использованием алгоритмов поиска и устранения неисправностей.

**Время выполнения работы:** 1 час 30 минут.

**Ход работы:**

1. Проведение преподавателем инструктажа по технике безопасности;
2. Изучить краткие теоретические сведения и законспектировать в технический отчет;
3. Последовательно выполнить порядок выполнения работы;
4. Письменно ответить на контрольные вопросы.

### Краткие теоретические сведения

Можно провести некоторые аналогии в строении роботов и людей. Блок управления можно сравнить с мозгом, двигатели соотнести с мышцами, а датчики – с органами чувств. Существует огромное количество датчиков различной сложности, точности и использующие разные физические законы. Объединяет их то, что с их помощью робот (точнее «мозг» робота) может получить информацию о состоянии объектов внешнего мира. Простейшим датчиком может служить обычная кнопка, а к более сложным датчикам можно отнести камеру или датчик расстояния. Применение датчика к мобильному роботу строго обязательно, поскольку при самостоятельном перемещении робот всегда находится в меняющейся среде. Робот-манипулятор зачастую может обойтись без них, поскольку постоянно находится на одном и том же месте и работает с одними и теми же объектами. Несмотря на это, во многих случаях без них не обойтись.

Платформа раздатчика комплекса Оптима-1 содержит световой датчик. Он передаёт в блок управления информацию об общей освещённости. Данный датчик может работать в двух режимах: первый позволяет оценить

внешнюю освещённость, при втором режиме рядом с датчиком загорается LED и оценивается то, насколько объект отражает свет.

С точки зрения программирования датчики представляют собой источник информации и подключенные к датчикам контакты должны быть настроены на ввод: `pinmode(5, INPUT)`; По аналогии с командой `digitalWrite(13,HIGH)` для считывания показаний датчика используется команда `analogRead(5)`; или `digitalRead(5)`;

Цифровое (digital) значение подразумевает однозначный результат: единица (логическая истина) или ноль (логическая ложь), аналоговое (analog) значение дробно или в определённом диапазоне (например, в Arduino от 0 до 1023). Цифровое чтение с аналогового датчика или аналоговое чтение с цифрового датчика может дать непредсказуемый результат.

Поскольку чтение с датчика целесообразно проводить постоянно, имеет смысл обновлять значение соответствующей переменной в функции `loop()`. Это значение может быть использована в других процедурах. Пример программы, использующей датчик освещённости приведён ниже.

```
#include<Adafruit_PWMServoDriver.h>
#include <Adafruit_TCS34725.h>
Adafruit_PWMServoDriverpwm = Adafruit_PWMServoDriver(); void
setup(){
  pwm.begin();
}
void loop(){
  int k = 1;
  int light = analogRead(5);
  pwm.setPWM(0,0,210);
  delay(k*light);
  pwm.setPWM(0,0,160);
  delay(k*light);
}
```

Изменяя коэффициент  $k$  можно менять частоту работы двигателя.

### Порядок выполнения работы

1. Перенесите программу в среду Arduino IDE. Опишите полученный результат.
2. Поменяйте освещение датчика при помощи фонарика или прикрыв его рукой.
3. Измените программу таким образом, чтобы в зависимости от общего освещения робот менял положение схвата.
4. Напишите программу, которая меняет степень сжатия схвата в зависимости от показания датчика освещения.
5. Включите светодиод около датчика. Опишите изменения в показании датчика при освещении фонариком или закрытии.

б. Реализуйте одновременное изменение скорости перемещения и угла поворота в зависимости от показания датчика.

### **Контрольные вопросы**

1. Приведите пример датчика с цифровым значением.
2. Что можно назвать датчиком в механических блоках управления?
3. Датчик освещения не возвращает нулевое значение даже в ночное время. Как Вы думаете, почему?
4. Разные датчики влияют на работу робота по-разному. Что необходимо сделать, чтобы при одинаковом уровне освещения, влияние на работу робота было одинаковым?
5. Предложите датчик, который мог бы существенно улучшить работу робота?
6. Датчики можно разделить на внешние и внутренние. Приведите примеры внутренних датчиков, которые могут быть использованы в робототехнике.



**ЛАБОРАТОРНАЯ РАБОТА №6**  
**«МИГАЕМ LED СВЕТОДИОДОМ. ТИПЫ ДАННЫХ. ЛОГИЧЕСКИЕ**  
**ОПЕРАТОРЫ. ОПЕРАТОР IF-ELSE, FOR-NEXT, WHILE, DO-WHILE.**  
**МИГАНИЕ LED СВЕТОДИОДОМ»**

**Цель работы:** при помощи операторов ветвления запрограммировать мигание светодиода с разной частотой.

**При выполнении лабораторной работы обучающийся должен:**

**Знать:** последовательность подключения светодиода и резистора в схеме; классификацию и виды отказов оборудования; технологическую последовательность разборки, ремонта и сборки узлов и механизмов мехатронных систем.

**Уметь:** читать электросхемы; кратко обосновывать и объяснить свои действия (текущие и планируемые); составлять план действия.

**Формируемые компетенции:**

ОК 01 Выбирать способы решения задач профессиональной деятельности, применительно к различным контекстам;

ОК 03 Планировать и реализовывать собственное профессиональное и личностное развитие;

ПК 2.1 Осуществлять техническое обслуживание компонентов и модулей мехатронных систем и мобильных робототехнических комплексов в соответствии с технической документацией.

**Время выполнения работы:** 1 час 30 минут.

**Ход работы:**

1. Проведение преподавателем инструктажа по технике безопасности;
2. Изучить краткие теоретические сведения и законспектировать в технический отчет;
3. Последовательно выполнить порядок выполнения работы;
4. Письменно ответить на контрольные вопросы.

**Краткие теоретические сведения**

Одним из фундаментальных понятий программирования является переменная. Она служит для хранения информации, в то время как операторы и функции проводят действия над переменными. Большая часть языков программирования имеет примерно одинаковый набор основных типов данных:

1. *Int* - это целый тип, который может хранить в себе только целые числа.
2. *Float* - данный тип является неточным. Он позволяет хранить не только целую часть, но, в отличие от типа *int*, и дробную.
3. *Double* - данный тип ничем не отличается от предыдущего в Arduino IDE.
4. *Char* - в данный тип данных можно записывать символы
5. *Bool* - хранит в себе значения логического типа: «правду» - *true*,

либо «ложь» - false. Использование данного типа позволяет в зависимости от текущей ситуации позволяет выбрать одно из двух действий.

Можно также создать массив любого типа. Стоит различать процесс создания (определения) переменной и процесс задания (присвоения значения) переменной. В первом случае указывается тип будущей переменной и её имя, а во втором случае переменная начинает ассоциироваться с конкретным значением.

Сложно найти задачу, в которой действия должны выполняться всегда по одному и тому же сценарию и не зависеть от актуальных условий. Для этого во всех языках программирования существуют логические операторы ветвления *if*. В конструкции оператора ветвления обязательно должно быть однозначное условие и должны быть описаны оба варианта исполнения программы в зависимости от значения логического выражения. В общем случае оператор ветвления выглядит так:

```
if( <логическое условие> ){
    <если условие истинно>
} else {
    <если условие ложно>
}
```

Помимо операторов ветвления фундаментальными также являются операторы цикла. Они позволяют выполнять одну часть программы либо заданное число раз, либо пока выполняется (или не выполняется) определённое условие. Если заранее известно, сколько будет циклов, то рекомендуется использовать оператор *for*, если же цикл должен завершаться по условию – оператор *while*.

```
for( int i = 0; I < 100; i++){<считать циклы от 0 до 100 с шагом в 1>
}
while( <логическое условие> ){
    <пока условие истинно>
}
do{
    <пока условие не выполнено>
} while( <логическое условие> )
```

Ветвление и циклы могут быть вложенными и входить в состав более сложных операторов и циклов. Программист всегда должен следить за тем, чтобы оба сценария ветвления были необходимы, а цикл не будет выполняться бесконечно. Помните, что программирование Arduino подразумевает бесконечный циклический вызов функции *loop()*, и очень часто без циклов можно обойтись.

### Порядок выполнения работы

1. Модифицируйте программу предыдущей лабораторной работы, чтобы при низком значении освещённости робот не совершал никаких действий.

2. Модифицируйте программу управления LED так, чтобы мерцание осуществлялось только в случае, когда светодиод горит дольше, чем не

горит.

3. Запретите перемещение робота при модуле суммы углов во второй и третьей оси большему, чем 120 градусов.

4. Запретите перемещение робота на угол, больший по модулю, чем 90 градусов.

5. Напишите программу управления мерцания светодиодам последовательно в трёх режимах: а) красный-желтый-зелёный по 1с; б) красный переходит в жёлтый затем переходит в зелёный и обратно в красный по 1с каждый переход; в) переливается произвольными цветами без резкой смены цвета.

### Контрольные вопросы

1. Приведите пример, в котором оператор ветвления не имеет альтернативного действия. Основное же действие по-прежнему выполняется только при логическом условии.

2. Почему не рекомендуется использовать цикл *while* с условием (*true*)?

3. В каком случае может понадобиться шаг счётчика цикла *for* отличный от единицы?

4. Приведите пример программы, где будет необходимо написать один цикл внутри другого.

5. Может ли сложиться такая ситуация, что тело цикла не будет выполнено ни разу? Если да, то приведите пример.

6. В каком случае есть смысл использовать для счётчика цикла другие типы данных?

7. В условии оператора *if* часто применяют пороговые значения, выше (или ниже) которых меняется исполняемая ветка программы. Как можно рассчитать такие значения?

## ЛАБОРАТОРНАЯ РАБОТА №7а «УПРАВЛЕНИЕ RGB LED СВЕТОДИОДОМ. УПРАВЛЯЮЩИЕ ОПЕРАТОРЫ SWITCH-CASE»

**Цель работы:** разработать алгоритм управления RGB светодиодом.

**При выполнении лабораторной работы обучающийся должен:**

**Знать:** правила техники безопасности при проведении работ по ремонту, техническому обслуживанию, контролю и испытаниям мехатронных систем; концепцию бережливого производства и классификацию и виды отказов оборудования.

**Уметь:** понимать общий смысл четко произнесенных высказываний на известные темы (профессиональные и бытовые); понимать тексты на базовые профессиональные темы; участвовать в диалогах на знакомые общие и профессиональные темы.

**Формируемые компетенции:** ОК1, ОК2, ПК 2.1.

ОК 01 Выбирать способы решения задач профессиональной деятельности, применительно к различным контекстам;

ОК 02 Осуществлять поиск, анализ и интерпретацию информации, необходимой для выполнения задач профессиональной деятельности;

ПК 2.1 Осуществлять техническое обслуживание компонентов и модулей мехатронных систем и мобильных робототехнических комплексов в соответствии с технической документацией.

**Время выполнения работы:** 1 час 30 минут.

**Ход работы:**

1. Проведение преподавателем инструктажа по технике безопасности;
2. Изучить краткие теоретические сведения и законспектировать в технический отчет;
3. Последовательно выполнить порядок выполнения работы;
4. Письменно ответить на контрольные вопросы.

### Краткие теоретические сведения

Одна из задач прикладного программиста сводится к написанию программы, удобной любому человеку. Прямое управление цветом RGB светодиода требует от обывателя знания схемы RGB и основ программирования. Было бы значительно удобнее, если программа получала бы номер цвета и, исходя из него, меняла бы яркость каждого компонента RGB светодиода. Для решения этой задачи понадобится не только познакомиться с новым оператором множественного ветвления, но и организовать коммуникацию блока управления с компьютером.

```
void setup(){  
  pinMode(11,OUTPUT);  
  pinMode(12,OUTPUT);  
  pinMode(13,OUTPUT);  
  Serial.start(9600);
```

```

}
void loop(){
int k = Serial.read();
if (k=='1') {
digitalWrite(11, HIGH);
digitalWrite(12, LOW);
digitalWrite(13, LOW);
} elseif (k=='2') {
...
}
...
}

```

Конструкция с несколькими подряд идущими операторами *if* будет работать, но есть большой шанс ошибиться, поскольку приходится писать очень похожие друг на друга команды и условия. Для таких случаев создан оператор множественного ветвления *switch*.

```

...
voidloop(){
int k = Serial.read();
switch (k) {
case '1':
digitalWrite(11, HIGH);
digitalWrite(12, LOW);digitalWrite(13, LOW);
break;
case '2':
digitalWrite(11, LOW);
digitalWrite(12, HIGH);
digitalWrite(13, LOW);
break;
...
}
...

```

В приведённом примере первый цвет будет красный, второй зелёный. *Serial* позволяет получать информацию от компьютера. Самый простой способ отправить плате символ – открыть монитор порта в Arduino IDE. Если в течение отведённого периода символ не будет получен, то значение *Serial.read()* будет равно (-1). *Serial.write()* используется для обратной связи с контроллера. Отправленные контроллером данные видны в окне монитора порта.

### Порядок выполнения работы

1. Напишите программу, которая отправляет обратно полученные от компьютера символы.
2. Напишите программу, которая в зависимости от присланного символа (числа) меняет цвет RGB светодиода. Нумерацию и описание цветов

используйте из предыдущей лабораторной работы.

3. Модифицируйте программу, чтобы она проверяла корректность полученного символа и при необходимости возвращала ошибку.

4. Попробуйте заменить операторы *digitalWrite()* на *analogWrite()*. Помните, что второе значение может быть строго в диапазоне от 0 до 1023.

### **Контрольные вопросы**

1. В каком случае может использоваться программа, пересылающая сообщение? В чем опасность повсеместного использования таких программ?

2. Почему при передаче данных между устройствами принято говорить о передаваемых строчках, а не о числах?

3. Что произойдёт, если в операторе *switch* убрать все команды прерывания *break*?

4. Почему аналоговое значение не может быть выше цифрового *HIGH*?

## ЛАБОРАТОРНАЯ РАБОТА №76 «ARDUINO IDE: ПРОГРАММИРОВАНИЕ СЕРВОДВИГАТЕЛЯ»

**Цель работы:** обучение программированию серводвигателей в среде Arduino IDE, управление перемещением манипулятора «Optima-1».

**При выполнении лабораторной работы обучающийся должен:**

**Знать:** правила техники безопасности при проведении работ по ремонту, техническому обслуживанию, контролю и испытаниям мехатронных систем; концепцию бережливого производства и классификацию и виды отказов оборудования.

**Уметь:** понимать общий смысл четко произнесенных высказываний на известные темы (профессиональные и бытовые); понимать тексты на базовые профессиональные темы; участвовать в диалогах на знакомые общие и профессиональные темы.

**Формируемые компетенции:**

ОК 02 Осуществлять поиск, анализ и интерпретацию информации, необходимой для выполнения задач профессиональной деятельности;

ОК 03 Планировать и реализовывать собственное профессиональное и личностное развитие;

ПК 2.3 Производить замену и ремонт компонентов и модулей мехатронных систем и мобильных робототехнических комплексов в соответствии с технической документацией.

**Время выполнения работы:** 1 час 30 минут.

**Ход работы:**

1. Проведение преподавателем инструктажа по технике безопасности;
2. Изучить краткие теоретические сведения и законспектировать в технический отчет;
3. Последовательно выполнить порядок выполнения работы;
4. Письменно ответить на контрольные вопросы.

### Краткие теоретические сведения

Для работы с серводвигателем в среде Arduino IDE предусмотрена библиотека Servo.h,

которая находится на диске, поставляемом с роботом или в свободном доступе по адресу <https://github.com/arduino-libraries/Servo/blob/master/src/Servo.h> Кратко рассмотрим основной смысл этой библиотеки. Серводвигатель активируется созданием объекта класса Servo и определением пинов командой attach().

Методы, которые используются в библиотеке Servo.h:

attach(pin) – связывает сервомотор с пином входа/выхода.

attach(pin, min, max) – связывает с пином, устанавливая минимальное и максимальное значение в микросекундах, по умолчанию – 544 и 2400 соответственно.

write() – устанавливает угол сервомотора в градусах.

writeMicroseconds() – устанавливает ширину пульсирования

серводвигателя в микросекундах.

`read()` – возвращает последнее записанное значение ширины пульсации в качестве угла от 0 до 180.

`readMicroseconds()` – возвращает последнее значение ширины пульсации в микросекундах.

`attached()` – возвращает `true`, если есть подключенный серводвигатель.

`detach()` – останавливает подключенные серводвигатели, отключает их от пинов входа/выхода.

### Начало работы

Рассмотрим вращение серводвигателя в стандартном примере Sweep.

Код программы:

Подключаем библиотеку Servo.h:

```
#include<Servo.h>
```

Создаем объект `myservo` класса `Servo` для управления сервомотором `Servomy servo`;

Переменная для хранения позиции серводвигателя `int pos = 0`;

```
Определяем 8-й пин как пин для серводвигателя void setup() {  
myservo.attach(8);  
}
```

```
void loop() {  
for (pos = 0; pos <= 180; pos += 1) {  
myservo.write(pos);  
delay(15);  
}  
for (pos = 180; pos >= 0; pos -= 1) {  
myservo.write(pos);  
delay(15);  
}  
}
```

Как и в случае с плавным затуханием светодиодов, здесь у нас есть 2 цикла – вращение по и против часовой стрелки. Задержка необходима, чтобы сервомотор успевал совершить движение.

### Порядок выполнения работы

1. Составить и проверить программу последовательного управления осями по следующему алгоритму (относительно базового положения: манипулятор направлен вверх на 180 град, 1-я ось в среднем положении):  
Ось №1: +90, Ось №2: -45, Ось №3: +45, Ось №4: +90, СТОП.

2. Составить и проверить программу одновременного управления осями из базового положения: Ось №2: -45, Ось №3: +45, СТОП.

### Контрольные вопросы

1. Что такое серводвигатель? В чем его плюсы и минусы?
2. Каким образом организовано взаимодействие серводвигателей со средой Arduino IDE? Что необходимо для корректной работы серводвигателей?
3. Сколько осей есть в «Optima-1»? Как они устроены?



**ЛАБОРАТОРНАЯ РАБОТА №8**  
**«ОПЕРАЦИИ ПО ПЕРЕМЕЩЕНИЮ МАНИПУЛЯТОРА.**  
**ЗАДАЧА ПРЯМОЙ КИНЕМАТИКИ»**

**Цель работы:** решить задачу прямой кинематики для робота-манипулятора Оптима-1. Написать программу для нахождения расстояния между начальным и конечным положением схвата.

**При выполнении лабораторной работы обучающийся должен:**

**Знать:** прямую и обратную задачи кинематики манипулятора; стандарты, положения, методические и другие нормативные материалы по аттестации, испытаниям, эксплуатации и ремонту оборудования мехатронных систем.

**Уметь:** понимать общий смысл четко произнесенных высказываний на известные темы (профессиональные и бытовые); понимать тексты на базовые профессиональные темы; участвовать в диалогах на знакомые общие и профессиональные темы;

**Формируемые компетенции:**

ОК 02 Осуществлять поиск, анализ и интерпретацию информации, необходимой для выполнения задач профессиональной деятельности;

ОК 04 Работать в коллективе и команде, эффективно взаимодействовать с коллегами, руководством, клиентами;

ПК 2.1 Осуществлять техническое обслуживание компонентов и модулей мехатронных систем и мобильных робототехнических комплексов в соответствии с технической документацией.

**Время выполнения работы:** 1 час 30 минут.

**Ход работы:**

1. Проведение преподавателем инструктажа по технике безопасности;
2. Изучить краткие теоретические сведения и законспектировать в технический отчет;
3. Последовательно выполнить порядок выполнения работы;
4. Письменно ответить на контрольные вопросы.

**Краткие теоретические сведения**

Согласованная работа всех двигателей робота-манипулятора является необходимым условием правильности выполнения задачи роботом. Большинство задач роботов-манипуляторов сводится к последовательному посещению точек пространства. Управлять же роботом напрямик в Декартовых координатах мы не можем, поскольку двигатели перемещают схват по своим осям вращения, расположение которых зависит от других осей. Одной из приоритетных задач при конструировании робота и программирования блока управления является решение задач прямой и обратной кинематики. При помощи функций прямой кинематики можно будет находить координаты инструмента в реальном мире, зная повороты в каждой оси. Обратная кинематика позволит рассчитать повороты всех осей, необходимые для достижения точки с известными координатами. Для

решения кинематики необходимо знать линейные размеры всех плеч. Обозначим их за  $l_n$ , где  $n$  – номер плеча.

Обратите внимание на рисунок осей около основания манипулятора. Ось  $X$  направлена от основания вдоль длинной стороны стола, ось  $Y$  вдоль короткой стороны стола, ось  $Z$  вертикально вверх. Для определения каждой координаты необходимо найти проекции каждого плеча робота на оси, а затем их сложить. Положение первого плеча неизменно. Второе плечо наклонено к оси  $Z$  под углом поворота второй оси робота. Третье плечо наклонено к оси  $Z$  под углом, равным сумме второго и третьего двигателя робота. Последнее плечо по аналогии представляет собой сумму поворотов во втором, третьем и четвёртом двигателях. Зная, что проекция равна произведению длины и косинуса угла наклона, запишем выражение

$$Z=l_1+c_2l_2+(c_2+c_3)l_3+(c_2+c_3+c_4)l_4$$

Соответствующие синусы покажут проекцию на плоскость стола.

$$W=s_2l_2+(s_2+s_3)l_3+(s_2+s_3+s_4)l_4$$

Рассмотрев плоскость стола легко заметить, что  $Y$  пропорциональна косинусу, а  $X$  синусу поворота первой оси.  $X=s_1*W$ ;  $Y=c_1*W$

Найденные значения и являются решением задачи прямой кинематики.

### Порядок выполнения работы

1. При помощи линейки измерьте плечи манипулятора. Измерения запишите в таблицу.
2. Реализуйте алгоритм расчёта координат инструмента (массив дробных чисел длиной 3) из углов поворота двигателей (массив дробных чисел длиной 4).
3. Проверьте работоспособность программы. Возьмите некоторые конфигурации робота, запустите программу, измерьте реальное положение схвата относительно основания. Измерения внесите в таблицу 1.

Таблица 1

Номер измерения	Конфигурация осей (градусы)	Рассчитанная координата	Измеренная координата
1	[45, -45, 60, -15]		
2			
3			

### Контрольные вопросы

1. Приведите пример устройств, в которых двигатели перемещают инструмент строго вдоль декартовых осей координат.
2. Можно ли решить задачу прямой кинематики другим способом?
3. \*Реализуйте проверку перед перемещением робота, находится ли конечная точка над поверхностью стола.
4. В данной работе были найдены только координаты инструмента. Имеет ли значение под каким углом находится. Как это можно рассчитать?
5. Что нужно делать, когда робот постоянно «не доходит» до заданной точки 2 см? Смещение всегда направлено в одну сторону.
6. Какие устройства кроме роботов требуют решения прямой кинематики?

**ЛАБОРАТОРНАЯ РАБОТА №9**  
**«ОПЕРАЦИИ ПО ПЕРЕМЕЩЕНИЮ МАНИПУЛЯТОРА.**  
**ЗАДАЧА ОБРАТНОЙ КИНЕМАТИКИ»**

**Цель работы:** решить задачу обратной кинематики по перемещению робота-манипулятора Оптима-1; написать функцию для перемещения инструмента на заданный вектор.

**При выполнении лабораторной работы обучающийся должен:**

**Знать:** прямую и обратную задачи кинематики манипулятора; стандарты, положения, методические и другие нормативные материалы по аттестации, испытаниям, эксплуатации и ремонту оборудования мехатронных систем.

**Уметь:** понимать общий смысл четко произнесенных высказываний на известные темы (профессиональные и бытовые); понимать тексты на базовые профессиональные темы; участвовать в диалогах на знакомые общие и профессиональные темы;

**Формируемые компетенции:**

ОК 01 Выбирать способы решения задач профессиональной деятельности, применительно к различным контекстам;

ОК 02 Осуществлять поиск, анализ и интерпретацию информации, необходимой для выполнения задач профессиональной деятельности;

ПК 2.1 Осуществлять техническое обслуживание компонентов и модулей мехатронных систем и мобильных робототехнических комплексов в соответствии с технической документацией.

**Время выполнения работы:** 1 час 30 минут.

**Ход работы:**

1. Проведение преподавателем инструктажа по технике безопасности;
2. Изучить краткие теоретические сведения и законспектировать в технический отчет;
3. Последовательно выполнить порядок выполнения работы;
4. Письменно ответить на контрольные вопросы.

**Краткие теоретические сведения**

Решение обратной задачи кинематики, как правило, значительно сложнее решения прямой. Для роботов с шестью осями общего алгебраического решения до сих пор не существует. Отсутствие универсального решения даёт большое количество способов решения. В данной лабораторной работе мы остановимся только на двух из них, о других же вы сможете узнать из курса «промышленной робототехники» на базе робота Оптима-2.

Самый очевидный вариант заключается в преобразовании формул прямой кинематики в систему уравнений, где  $X$ ,  $Y$ ,  $Z$  будут известны изначально. Найти в данном случае будет необходимо углы поворота.

$$Z=l1+ c2l2+(c2+c3)l3+ (c2+c3+c4)l4$$

$$W=s2l2+(s2+s3)l3+ (s2+s3+s4)l4$$

$$W = \sqrt{X^2 + Y^2}$$

Данная система не имеет точного решения. Для уточнения необходимо также задать необходимые значения наклона инструмента. Полное решение достаточно сложно и не будет изложено в лабораторной работе.

Второй подход связан с геометрическим решением. Решение также достаточно сложно, и поэтому будет только описан его ход. Зная координаты XY можно найти тангенс угла поворота первой оси робота, а вслед за ними сам угол. Также можно найти разницу высот инструмента и точки вращения второй оси. Также можно найти расстояние между этой точкой и инструментом. Таким образом, в треугольнике с вершинами в точках вращения осей 2, 3, 4 известны все стороны. Следовательно, можно использовать теорему косинусов для нахождения всех углов.

Решение задачи прямой и обратной кинематики могут быть реализованы лишь на мощных микроконтроллерах, поскольку требуют больших вычислительных ресурсов. Стоит отметить, что обратная задача очень часто даёт несколько решений. Выбор необходимого решения определяется условиями безопасности, быстроты и качества исполнения.

#### Порядок выполнения работы

1. Используя готовую функцию обратной кинематики, проведите эксперименты и заполните таблицу 1.

Таблица 1

Номер измерения	Координаты	Конфигурация осей посчитанная	Конфигурация осей измеренная
1	[12, -13.5, 30, -15]		
2			
3			

2. Сравните точность работы функции прямой и обратной кинематики.

3. Напишите программу по перемещению манипулятора последовательно по 5 удалённым друг от друга точкам.

4. Реализуйте возможность регулировки скорости перемещения в Вашей программе.

#### Контрольные вопросы

1. Основываясь на свойствах прямых и обратных тригонометрических функций, объясните с точки зрения алгебры почему у обратной задачи кинематики несколько решений.

2. Предложите другие способы решения обратной задачи кинематики.

3. Достаточно ли условие о нахождении инструмента над поверхностью стола для предотвращения столкновений?

4. Как Вы думаете, почему решение обратной задачи кинематики важно для применения робота на практике, в производстве?

5. Для проверки работоспособности функций обратной кинематики обычно используют функцию прямой кинематики. Почему?

**ЛАБОРАТОРНАЯ РАБОТА №10а**  
**«ПРОСТЫЕ ОПЕРАЦИИ ПО ПЕРЕМЕЩЕНИЮ ОБЪЕКТОВ В**  
**ПРОСТРАНСТВЕ С ПОМОЩЬЮ МАНИПУЛЯТОРА»**

**Цель работы:** реализовать алгоритм перемещения объектов в рабочем пространстве робота.

**При выполнении лабораторной работы обучающийся должен:**

**Знать:** простые операции по перемещению объектов в пространстве с помощью манипулятора; программирование оптодатчика.

**Уметь:** программировать оптодатчик; обеспечивать требуемое перемещение объекта в пространстве с заданной ориентацией.

**Формируемые компетенции:**

ОК 01 Выбирать способы решения задач профессиональной деятельности, применительно к различным контекстам;

ОК 03 Планировать и реализовывать собственное профессиональное и личностное развитие;

ПК 2.2 Диагностировать неисправности мехатронных систем и мобильных робототехнических комплексов с использованием алгоритмов поиска и устранения неисправностей.

**Время выполнения работы:** 1 час 30 минут.

**Ход работы:**

1. Проведение преподавателем инструктажа по технике безопасности;
2. Изучить краткие теоретические сведения и законспектировать в технический отчет;
3. Последовательно выполнить порядок выполнения работы;
4. Письменно ответить на контрольные вопросы.

**Краткие теоретические сведения**

Задачу по перемещению объектов внутри рабочего пространства робота можно разделить на несколько этапов: перемещение схвата к объекту (шайбе), захват, перемещение в конечную точку, разжатие. Сложность реализации алгоритма заключается в том, что перемещение необходимо делать синхронно, и только после достижения начальной точки начинать движение схвата. Для большего удобства программирования и уменьшения вероятности ошибки, часто используемые действия целесообразно выносить в отдельные функции. Синтаксис функции представлен ниже:

```
intfunctionName (intarg1, intarg2){  
...  
return answer;  
}
```

В первую очередь, указывается тип возвращаемого значения, затем имя функции. В скобках при наличии указываются аргументы, а в фигурных скобках реализуется сам алгоритм. После завершения всех расчётов результат возвращается в основную программу при помощи команды *return*.

Для перемещения объектов понадобятся две функции для открытия и закрытия схвата. Для этого внутри функции нужно использовать данные калибровки робота. Функции перемещения должны использовать решение обратной задачи кинематики для нахождения нужных значений углов, после чего перемещать все двигатели в нужные положения.

### **Порядок выполнения работы**

1. Напишите функцию, которая открывает схват и закрывает его через несколько секунд. Манипулятор при этом должен оставаться неподвижным.

2. Вынесите алгоритм работы с инструментом в отдельные функции. Повторно проверьте работоспособность программы.

3. Используя готовое решение обратной кинематики, напишите функцию для перемещения в заданную точку. Работа двигателей при этом должна быть синхронной.

4. Запрограммируйте робота на перемещение шашки из произвольной точки зелёной области в произвольную точку красной. Попробуйте использовать данную программу на другом роботе Оптима-1.

### **Контрольные вопросы**

1. Как Вы думаете, почему функциям обычно стараются давать названия в виде глаголов?

2. Переменные, объявленные до функции *setup()* могут использоваться в программе повсеместно. Как их можно применить при выборе нужного решения задачи обратной кинематики?

3. Какой датчик необходимо установить в схват, чтобы можно было хватать объекты разных размеров?

4. Можно ли вызвать функцию *setup()* из функции *loop()*?

5. К каким последствиям может привести вызов функцией самой себя? Всегда ли можно вместо этого использовать циклы?

6. Предложите способ хранения информации о количестве перенесённых шашек.

**ЛАБОРАТОРНАЯ РАБОТА №106**  
**«ARDUINO IDE: ПРОСТЫЕ ОПЕРАЦИИ ПО ПЕРЕМЕЩЕНИЮ**  
**ОБЪЕКТОВ В ПРОСТРАНСТВЕ С ПОМОЩЬЮ МАНИПУЛЯТОРА»**

**Цель работы:** обучение программированию серводвигателей в среде Arduino IDE, управление перемещением манипулятора «Optima-1».

**При выполнении лабораторной работы обучающийся должен:**

**Знать:** простые операции по перемещению объектов в пространстве с помощью манипулятора; программирование оптодатчика.

**Уметь:** программировать оптодатчик; обеспечивать требуемое перемещение объекта в пространстве с заданной ориентацией.

**Формируемые компетенции:**

ОК 02 Осуществлять поиск, анализ и интерпретацию информации, необходимой для выполнения задач профессиональной деятельности;

ОК 03 Планировать и реализовывать собственное профессиональное и личностное развитие;

ПК 2.1 Осуществлять техническое обслуживание компонентов и модулей мехатронных систем и мобильных робототехнических комплексов в соответствии с технической документацией.

**Время выполнения работы:** 1 час 30 минут.

**Ход работы:**

1. Проведение преподавателем инструктажа по технике безопасности;
2. Изучить краткие теоретические сведения и законспектировать в технический отчет;
3. Последовательно выполнить порядок выполнения работы;
4. Письменно ответить на контрольные вопросы.

**Краткие теоретические сведения**

Теоретическая механика – наука об общих законах механических взаимодействий между материальными телами, а также об общих законах движения тел по отношению друг к другу.

Кинематика – раздел теоретической механики, в котором изучается движение механических систем с геометрической точки зрения, без учета причин (сил), вызывающих это движение и изменение движения.

Кинематическая пара – соединение двух соприкасающихся звеньев, допускающее их относительное движение.

Кинематическая цепь – это система звеньев, которые связаны между собой кинематическими парами.

Кинематическая схема – это графическая схема отображения рабочих узлов и блоков механизма конструкции.

С математическим понятием «система координат» тесно связано техническое понятие «степень подвижности» механической системы, в частности манипулятора. С его помощью определяется, сколько независимых движений должен совершать манипулятор или отдельное его звено, чтобы

оказаться в нужной точке. Например, для того чтобы захват робота, работающего в прямоугольной системе координат, попал в определенную точку плоскости, он должен переместиться на определенное расстояние по оси X, а затем — по оси Y. Два перемещения по двум осям соответствуют двум степеням подвижности.

Свободное тело в пространстве обладает шестью степенями свободы. Любой предмет можно перемещать в трех взаимно перпендикулярных направлениях, т. е. вдоль осей X, Y Z, и поворачивать его относительно этих осей. Следовательно, теоретически каждое из сочленений звена может иметь шесть степеней подвижности.

На практике число степеней подвижности в каждом сочленении звена обычно не превышает одной, что в первую очередь объясняется простотой реализации такой конструкции.

Тип применяемой в конструкции робота системы координат резко влияет на размер зоны обслуживания, которая при замене прямоугольной системы координат на цилиндрическую увеличивается в 9,6 раза, а при использовании полярной системы координат — в 29,7 раза. Помимо системы координат на размеры зоны обслуживания (или рабочего пространства) влияет также число степеней подвижности робота

Именно с разработки структурной схемы манипулятора, системы координат, в которой он должен будет работать, определения зоны обслуживания «мертвого» пространства начинается работа по созданию конструкции робота.

### **Начало работы**

Рассмотрим вращение серводвигателя в стандартном примере Sweep. Код программы:

Подключаем библиотеку Servo.h:

```
#include<Servo.h>
```

Создаем объект myservo класса Servo для управления сервомотором  
Servomy servo;

```
Переменная для хранения позиции серводвигателя int pos = 0;
```

```
Определяем 8-й пин как пин для серводвигателя void setup() {
```

```
myservo.attach(8);
```

```
}
```

```
void loop() {
```

```
for (pos = 0; pos<= 180; pos += 1) {
```

```
myservo.write(pos);
```

```
delay(15);
```

```
}
```

```
for (pos = 180; pos>= 0; pos -= 1) {
```

```
myservo.write(pos);/
```

```
delay(15);
```

```
}
```

```
}
```



Как и в случае с плавным затуханием светодиодов, здесь у нас есть 2 цикла – вращение по и против часовой стрелки. Задержка необходима, чтобы сервомотор успевал совершить движение.

### **Порядок выполнения работы**

1 Составить и проверить программу последовательного управления осями по следующему алгоритму (относительно базового положения: манипулятор направлен вверх на 180 град., 1-я ось в среднем положении):  
Ось №1: +90, Ось №2: -45, Ось №3: +45, Ось №4: +90, СТОП.

2 Составить и проверить программу одновременного управления осями из базового положения: Ось №2: -45, Ось №3: +45, СТОП.

### **Контрольные вопросы**

- 1 Что такое серводвигатель? В чем его плюсы и минусы?
- 2 Каким образом организовано взаимодействие серводвигателей со средой Arduino IDE? Что необходимо для корректной работы серводвигателей?
- 3 Сколько осей есть в «Optima-1»? Как они устроены?

**ЛАБОРАТОРНАЯ РАБОТА №11-12**  
**«ИЗУЧЕНИЕ ПРИНЦИПОВ РАБОТЫ RGB ДАТЧИКА**  
**РАСПОЗНАВАНИЯ ЦВЕТА. ПРОГРАММИРОВАНИЕ**  
**ОПТИЧЕСКОГО ДАТЧИКА. ПРИМЕНЕНИЕ ОПТИЧЕСКОГО**  
**ДАТЧИКА В АЛГОРИТМАХ УПРАВЛЕНИЯ РОБОТОМ-**  
**МАНИПУЛЯТОРОМ»**

**Цель работы:** запрограммировать робота для работы с датчиком света.

**При выполнении лабораторной работы обучающийся должен:**

**Знать:** алгоритмы управления роботом – манипулятором; длину волны датчиков цвета.

**Уметь:** подключать датчик цвета; определять цвет/уровень освещенности с помощью соответствующих фотодиодов.

**Формируемые компетенции:**

ОК 01 Выбирать способы решения задач профессиональной деятельности, применительно к различным контекстам;

ОК 02 Осуществлять поиск, анализ и интерпретацию информации, необходимой для выполнения задач профессиональной деятельности;

ПК 2.3 Производить замену и ремонт компонентов и модулей мехатронных систем и мобильных робототехнических комплексов в соответствии с технической документацией.

**Время выполнения работы:** 3 часа 00 минут.

**Ход работы:**

1. Проведение преподавателем инструктажа по технике безопасности;
2. Изучить краткие теоретические сведения и законспектировать в технический отчет;
3. Последовательно выполнить порядок выполнения работы;
4. Письменно ответить на контрольные вопросы.

**Краткие теоретические сведения**

Вы уже знакомы с общими принципами работы датчиков. Оптический датчик несколько сложнее одноканальных, поскольку состоит, по аналогии со светодиодом, из трёх составляющих. Следовательно, помимо питания нужно подключать три разных контакта для красного, зелёного и синего каналов. Каждый из каналов поотдельности не может дать целостное представление о цвете объекта. Также показания очень сильно зависят от уровня освещения и гладкости поверхности. Чтобы избежать неверного определения цвета необходимо выделить отдельные признаки.

Природа цветов не позволяет одинаково воспринимать все каналы. Например, зелёный цвет встречается повсеместно, а красный канал редко бывает выше 150 из 255. Существуют другие цветовые схемы, позволяющие сгладить эту разницу. Кроме того, глянцевые поверхности могут по-разному отражать разный свет. Несмотря на это, оптический датчик достаточно точный при должном алгоритме.

### Порядок выполнения работы

1. Напишите программу, которая раз в секунду отправляет по серийному порту показания оптического датчика. Для коммуникации используйте *Serial*.

2. Поочерёдно установите на датчик все шашки зелёного и красного цвета. Возвращаемые результаты запишите в таблицу 1. Каждый раз проверяйте общий уровень освещенности при отсутствии шашки.

Таблица 1

Цвет и номер шашки	Красный	Зеленый	Синий	Уровень освещения
Красная 1				
Зеленая 2				
Зеленая 3				
.....				

3. Повторите эксперимент с другим освещением (включите или выключите часть освещения комнаты, закройте или откройте шторы).

4. Предложите алгоритм определения цвета шашки при любом освещении.

5. Заполните таблицу 2 кандидатов условий. Предложите свои критерии оценки цвета

Таблица 2

Цвет и номер шашки	Разница красного и зеленого	Отношение красного и зеленого
Красная 1		
.....		

6. Напишите программу, по которой можно определить цвет шашки над оптодатчиком.

7. Модифицируйте программу предыдущей лабораторной работы, чтобы в зависимости от цвета шашки над оптодатчиком менялся угол захвата.

8. Заставьте робота взять шашку с датчика и переместить на поверхностьстола.

### Контрольные вопросы

1. Нужно ли использовать синий для оценки цвета шашки?
2. Можно ли использовать только один канал оптодатчика для определения цвета?
3. Как сравнить яркость света при разных цветах. Достаточно ли найти сумму компонентов?
4. Разные датчики могут определять цвет по-разному. Как убрать эти расхождения?
5. Почему в современных камерах зелёных датчиков 50%, а красных и синих по 25%?

**ЛАБОРАТОРНАЯ РАБОТА №13**  
**«ВЗАИМОДЕЙСТВИЕ РОБОТА И ДАТЧИКОВ.**  
**ПРОГРАММИРОВАНИЕ СОРТИРОВОЧНОЙ УСТАНОВКИ»**

**Цель работы:** запрограммировать робот-манипулятор Оптима-1 для сортировки шашек равного цвета.

**При выполнении лабораторной работы обучающийся должен:**

**Знать:** устройство сортировочной установки; принципы взаимодействия роботизированной установки и датчиков.

**Уметь:** выделять наиболее значимое в перечне информации; оценивать практическую значимость результатов поиска; оформлять результаты поиска.

**Формируемые компетенции:**

ОК 01 Выбирать способы решения задач профессиональной деятельности, применительно к различным контекстам;

ОК 03 Планировать и реализовывать собственное профессиональное и личностное развитие;

ПК 2.2 Диагностировать неисправности мехатронных систем и мобильных робототехнических комплексов с использованием алгоритмов поиска и устранения неисправностей.

**Время выполнения работы:** 1 час 30 минут.

**Ход работы:**

1. Проведение преподавателем инструктажа по технике безопасности;
2. Изучить краткие теоретические сведения и законспектировать в технический отчет;
3. Последовательно выполнить порядок выполнения работы;
4. Письменно ответить на контрольные вопросы.

**Краткие теоретические сведения**

Автоматическая сортировка – один из наиболее распространённых процессов, выполняемых роботами. Сортировочные системы установлены на крупных и мелких складах, почтовых центрах, производственных комплексах, при контроле качества продукции. Задача сортировки всегда сводится к трём подзадачам: оценка, перемещение объекта, возврат.

Существует общий принцип программирования роботов: оцени, обдумай, сделай. Большая часть реальных роботов работают по этому алгоритму. В нашем случае оценка означает определение цвета шашки. Обдумывание подразумевает перенос шашки на незанятую позицию в нужном секторе. Действие подразумевает непосредственно перенос.

Блок управления должен принимать решения не только по показаниям датчиков, но и по конфигурации самого робота. Если нет внутренних датчиков поворота осей, то для этого может быть использована модель робота и стола. Модель робота хранит рассчитанное текущее состояние двигателей, а модель стола будет хранить положение шашек. Моделирование – мощный инструмент предсказания состояния системы, в том числе и в

будущем. Так можно предотвратить очень высокую скорость движения робота или столкновение с движущимися объектами.

### **Порядок выполнения работы**

1. Создайте в программе массив, который будет хранить информацию о наличии или отсутствии шашки в ячейке сектора. После переноса шашки в данную ячейку, она должна отмечаться как занятая.
2. Реализуйте алгоритм сортировки. Укажите в комментариях часть программы, отвечающие за оценку, анализ и действие.
3. Проверьте работу алгоритма. Какой процент ошибок. Повторите эксперимент не менее 20 раз.
4. Запрограммируйте работу раздатчика, оформите вынос новой шашки в отдельную функцию.
5. Напишите программу для сортировки по секторам всех шашек, расположенных в раздатчике. Оцените точность перемещения и процент ошибок. Предложите пути их устранения.

### **Контрольные вопросы**

1. Приведите пример сортировочной установки при проверке качества изделия. Какие датчики должны использоваться в данной установке.
2. Если используются несколько датчиков, и они дают разный результат, каковы могут быть принципы работы алгоритма?
3. В какой момент целесообразно осуществлять проверку: в момент захвата или в момент выхода шайбы? Почему?
4. Можно ли определить отсутствие шашки? Как?
5. Почему при оценке цвета стараются не использовать каналы RGB и переводят в другие цветовые схемы?

**ЛАБОРАТОРНАЯ РАБОТА №14**  
**«ПРОГРАММИРОВАНИЕ ДАТЧИКА ВЛАЖНОСТИ. ПРИМЕНЕНИЕ**  
**ДАТЧИКА ВЛАЖНОСТИ В АЛГОРИТМАХ УПРАВЛЕНИЯ**  
**РОБОТОМ-МАНИПУЛЯТОРОМ»**

**Цель работы:** познакомиться с принципами работы других датчиков, написать программу для работы с датчиком влажности.

**При выполнении лабораторной работы обучающийся должен:**

**Знать:** устройство датчика влажности; принципы взаимодействия роботизированной установки и датчиков влажности.

**Уметь:** выделять наиболее значимое в перечне информации; оценивать практическую значимость результатов поиска; оформлять результаты поиска.

**Формируемые компетенции:**

ОК 01 Выбирать способы решения задач профессиональной деятельности, применительно к различным контекстам;

ОК 04 Работать в коллективе и команде, эффективно взаимодействовать с коллегами, руководством, клиентами;

ПК 2.3 Производить замену и ремонт компонентов и модулей мехатронных систем и мобильных робототехнических комплексов в соответствии с технической документацией.

**Время выполнения работы:** 1 час 30 минут.

**Ход работы:**

1. Проведение преподавателем инструктажа по технике безопасности;
2. Изучить краткие теоретические сведения и законспектировать в технический отчет;
3. Последовательно выполнить порядок выполнения работы;
4. Письменно ответить на контрольные вопросы.

**Краткие теоретические сведения**

Существует несколько разных классификаций датчиков. Одна из них основана на том, получает ли датчик данные о координатах объекта или о состоянии в целом. Датчик GPS, камера, датчик расстояния и другие помогают ориентироваться роботу; датчик заряда батареи, влажности, давления, температуры, освещённости и др. определяют глобальные для робота состояния среды.

Датчик влажности может служить для контроля идеального состояния среды в помещении для работы роботов. В нашем случае его можно использовать для определения действий человека. В помещении, где есть люди, влажность обычно выше. Если робот используется для погрузки продуктов, то влажность влияет на их срок годности и жесткость. Последнее применение датчиков температуры и влажности – постоянная калибровка других точных датчиков, зависящих от температуры и влажности.

Влажность воздуха бывает абсолютная и относительная. Абсолютная показывает массовую долю паров воды в воздухе. Относительная влажность

учитывает температуру и давление и показывает процент паров воды от максимально возможного, выше которого начнут образовываться капельки тумана или дождя.

### **Порядок выполнения работы**

1. Методом проб и ошибок определите, к какому контакту подключен датчик влажности. Физически он находится на плате над Arduino UNO.
2. Реализуйте алгоритм остановки выполнения программы при очень низком или очень высоком показании датчика влажности.
3. Добавьте в программу вывод информации о влажности при запуске. Реализуйте опрос датчика и отправку на компьютер полученных данных в отдельной функции.
4. Перед окончанием занятия запишите показания датчика и сравните его с данными в начале следующего урока.

### **Контрольные вопросы**

1. Датчик влажности в некоторых странах устанавливают в стадионах для проверки работы обслуживающего персонала. Как Вы думаете, как должен работать алгоритм определения, была ли влажная уборка?
2. Калибровка датчика влажности обычно производится одновременно с датчиком температуры. Почему?
3. Датчик влажности может использовать разные физические явления. Самыми популярными являются устройства, измеряющие электропроводимость или оптическую проницаемость. Попробуйте объяснить принцип их работы.
4. Какие датчики могут быть использованы дополнительно для контроля условий работы робота или человека? Какие показатели они могут измерить?

**ЛАБОРАТОРНАЯ РАБОТА №15**  
**«ПРОГРАММИРОВАНИЕ ПОТЕНЦИОМЕТРА. ПРИМЕНЕНИЕ**  
**ПОТЕНЦИОМЕТРА В АЛГОРИТМАХ УПРАВЛЕНИЯ РОБОТОМ-**  
**МАНИПУЛЯТОРОМ»**

**Цель работы:** реализовать алгоритм управления скоростью исполнения движения робота по данным потенциометра.

**При выполнении лабораторной работы обучающийся должен:**

**Знать:** устройство и назначение потенциометра; принципы взаимодействия роботизированной установки и потенциометра.

**Уметь:** программировать потенциометр, применять потенциометр в роботизированных установках.

**Формируемые компетенции:**

ОК 01 Выбирать способы решения задач профессиональной деятельности, применительно к различным контекстам;

ОК 03 Планировать и реализовывать собственное профессиональное и личностное развитие;

ПК 2.1 Осуществлять техническое обслуживание компонентов и модулей мехатронных систем и мобильных робототехнических комплексов в соответствии с технической документацией.

**Время выполнения работы:** 1 час 30 минут.

**Ход работы:**

1. Проведение преподавателем инструктажа по технике безопасности;
2. Изучить краткие теоретические сведения и законспектировать в технический отчет;
3. Последовательно выполнить порядок выполнения работы;
4. Письменно ответить на контрольные вопросы.

**Краткие теоретические сведения**

Потенциометром (или потенциометрическим датчиком) называется устройство, определяющее положение контакта относительно катушки. Линейный потенциометр показывает расстояние, на которое сместился контакт, угловой – показывает угол поворота. Потенциометры используются для настройки приборов, быстрой корректировки, управления током и т.д. Классическим примером является регулятор громкости: датчик определяет, в каком положении находится реле и меняет выходной ток. Как результат – устанавливается определённая громкость звука динамика.

Робот-манипулятор Оптима-1 содержит один угловой потенциометр, который можно повернуть при помощи отвертки. Над потенциометром в столе есть специальное отверстие.

**Порядок выполнения работы**

1. Напишите функцию, которая считывает показания потенциометра и поворачивает вторую ось манипулятора на угол в диапазоне от -90 до +90 градусов. Максимальное значение потенциометра должно соответствовать углу в 90 градусов, минимальное значение – в -90 градусов.



2. Реализуйте алгоритм управления одной из осей в реальном времени при помощи потенциометра.
3. Используйте потенциометр для калибровки осей робота. Для этого выводите текущие значения, передаваемые в функцию *setPWM* компьютеру. Переключение между управляемыми осями должно осуществляться путем отправки номера оси от компьютера к плате.
4. Откалибруйте робота ещё раз и сравните результаты с предыдущими.

### **Контрольные вопросы**

1. В каком случае имеет смысл использовать несколько потенциометров для регулировки одного и того же выхода? Поможет ли это в увеличении диапазона или точности регулирования?
2. При решении каких задач из прежних лабораторных работ можно использовать потенциометр?
3. Приведите пример использования потенциометров не для ручной регулировки.
4. Предположим, что потенциометр возвращает максимальное значение 1023 при повороте не на максимальный угол. Предложите способ решения проблемы, чтобы датчик работал во всём диапазоне.
5. Что будет, если подключить потенциометр к светодиоду последовательно или параллельно? Ответ поясните.

## **ПРАВИЛА ТЕХНИКИ БЕЗОПАСНОСТИ ПРИ РАБОТЕ ОБУЧАЮЩИХСЯ В ЛАБОРАТОРИЯХ**

1. В целях предупреждения несчастных случаев напряжение выше 36 В следует считать опасным для жизни.

2. Начиная работу, обучающиеся должны убедиться в том, что на лабораторном столе (вводные рубильники, пакетные выключатели) нет напряжения.

3. Составление, разборку или изменение схемы производят только с разрешения преподавателя.

4. Запрещается включать вновь составленную или измененную схему без предварительной проверки ее преподавателем.

5. Запрещается прикасаться к токоведущим частям и металлическим частям незаземленных электрических аппаратов, если на щите имеется напряжение.

6. Все операции производить только одной рукой. При этом следует остерегаться прикосновений какой-либо частью тела к окружающим металлическим либо влажным предметам. Опасно, например, прикасаться одновременно к электрическим машинам, корпусу щита, водопроводным трубам, трубам центрального отопления или находиться на мокром либо цементном полу.

7. Перед включением напряжения следует убедиться в том, что все регулирующие аппараты находятся в исходном положении. После отключения напряжения необходимо немедленно восстановить на всех регулировочных аппаратах исходное положение.

8. Перед включением напряжения следует предупредить об этом всех участников работы. Необходимо убедиться, что никому из них не угрожает опасность попасть под напряжение.

9. Если при прикосновении к какой-либо части оборудования ощущается напряжение, то необходимо прекратить работу, выключить ток и вызвать преподавателя.

10. Если до начала работы или в ходе работы обнаружена неисправность оборудования, следует прекратить работу, отключить напряжение и сообщить преподавателю или инженеру о неполадках в работе. Устранять неполадки собственными силами запрещается.

11. При работе с цепями переменного тока, содержащими конденсаторы, следует соблюдать особую осторожность, имея в виду возможность значительного возрастания напряжения на отдельных участках по сравнению с напряжением источника тока вследствие возможного явления резонанса напряжений.

12. Запрещается прикасаться к приборам, находящимся на задней стенке щитов.

13. Запрещается обучающимся заходить за лабораторный стенд.

15. При необходимости пользоваться кнопкой аварийного срочного отключения.

16. Запрещается приступать к выполнению работы до тех пор, пока преподавателем не будет установлено, что студенту известны цель работы, метод

Ее выполнения, способ обращения с оборудованием, диапазон переменных величин и предполагаемые результаты.

17. Запрещается покидать лабораторию без разрешения преподавателя.

18. Запрещается оставлять без надзора установки, приведенные в рабочее состояние.

19. Перед началом работы следует распределить между членами бригады обязанности с таким расчетом, чтобы обеспечить соблюдение правил техники безопасности. Необходимо, например, обеспечить возможность быстрого аварийного отключения установки студентом, постоянно находящимся вблизи главного рубильника.

20. Рекомендуется останавливать агрегат всякий раз, когда возникает необходимость обсудить дальнейший план работы.

21. Запрещается переносить приборы с одного места на другое.

22. Запрещается трогать оборудование, неиспользуемое в данной работе.

23. После проработки настоящей инструкции на первом лабораторном занятии все обучающиеся обязаны расписаться в лабораторном журнале по технике безопасности.

24. Без инструктажа и отметки в журнале преподавателю категорически запрещается допускать обучающихся к лабораторным работам.

## КРИТЕРИИ ОЦЕНКИ ВЫПОЛНЕНИЯ ЛАБОРАТОРНОЙ РАБОТЫ

*Оценка «отлично» ставится в том случае, если обучающийся:*

- а) выполнил работу в полном объеме с соблюдением необходимой последовательности проведения опытов и измерений;
- б) самостоятельно и рационально выбрал и подготовил для опыта необходимое оборудование, все опыты провел в условиях и режимах, обеспечивающих получение результатов и выводов с наибольшей точностью;
- в) в представленном отчете правильно и аккуратно выполнил все записи, таблицы, рисунки, чертежи, графики, вычисления и сделал выводы;
- г) правильно выполнил анализ погрешностей;
- д) соблюдал требования безопасности труда.

*Оценка «хорошо» ставится в том случае, если выполнены требования к оценке «5», но:*

- а) опыт проводился в условиях, не обеспечивающих достаточной точности измерения,
- б) было допущено два-три недочета, или не более одной негрубой ошибки и одного недочета.

*Оценка «удовлетворительно» ставится, если:*

работа выполнена не полностью, но объем выполненной части таков, что позволяет получить правильные результаты и выводы, или если в ходе проведения опыта и измерений были допущены следующие ошибки:

- а) опыт проводился в нерациональных условиях, что привело к получению результатов с большей погрешностью,
- б), или в отчете были допущены в общей сложности не более двух ошибок (в записях единиц, измерениях, в вычислениях, графиках, таблицах, схемах, анализе погрешностей и т. д.), не принципиального для данной работы характера, но повлиявших на результат выполнения,
- в) или работа выполнена не полностью, однако объем выполненной части таков, что позволяет получить правильные результаты и выводы по основным, принципиально важным задачам работы.

*Оценка «неудовлетворительно» ставится в том случае, если:*

- а) работа выполнена не полностью, и объем выполненной части работы не позволяет сделать правильных выводов,
- б) или опыты, измерения, вычисления, наблюдения производились неправильно,
- в) или в ходе работы и в отчете обнаружилось в совокупности все недостатки, отмеченные в требованиях к оценке «3 - удовлетворительно».

## БИБЛИОГРАФИЧЕСКИЙ СПИСОК

1. Акимова, Н.А. Монтаж, техническая эксплуатация и ремонт электрического и электромеханического оборудования: учебник / Н.А. Акимова, Н.Ф. Котеленец, Н.И. Сентюрихин - М.: ОИЦ «Академия», 2013. - 411с.
2. Банци, М. Arduino для начинающих / М. Банци. - М.: Рид Групп, 2012. - 128 с.
3. Блум, Д. Изучаем Arduino. Инструменты и методы технического мастерства / Д. Блум - Издательство: БХВ-Петербург, 2015 – 334с.
4. Бокселл, Д. Изучаем Arduino. 65 проектов своими руками /Д. Бокселл - СПб.: Питер, 2017. - 400 с.
5. Быков, А.В. Компьютерные чертежно-графические системы для разработки конструкторской и технологической документации в машиностроении. Учебное пособие для нач. проф. образования / А.В. Быков, В.Н. Гаврилов, Л.М. Рыжкова, В.Я. Фадеев, под общей редакцией Л.А. Чемпинского - М.: Издательский центр "Академия", 2012. – 531с.
6. Васильков, А.В. Источники электропитания: Учебное пособие / А.В. Васильков, И. А. Васильков - М.: Форум, 2016. - 400 с.
7. Петин, В.А. Проекты с использованием контроллера Arduino / В.А. Петин - СПб.: БХВ-Петербург, 2014. - 400 с.
8. Таратынов, О.В., Технология машиностроения. Основы проектирования на ЭВМ: учеб. пособие / О.В. Таратынов, В.В. Клепиков, Б.М. Базров. - М.: ФОРУМ, 2017. - 608 с.
9. Карташов Г.Б., Дмитриев А.В. Основы работы на станках с ЧПУ / Г.Б. Карташов, А.В. Дмитриев – М.: Дидактические системы, 2012. – 158 с.
10. Ключев, А.С. Монтаж средств измерений и автоматизации: справочник / А.С. Ключев – М.: Энергоатомиздат, 2012 – 114 с.
11. Лоторейчук, Е.А. Теоретические основы электротехники: учебник / Е.А. Лоторейчук. - М.: ИД «ФОРУМ»: ИНФРА-М, 2017. - 317 с.
12. Приказ Минобрнауки России от 09.12.2016 № 1550 «Об утверждении федерального государственного образовательного стандарта среднего профессионального образования по специальности 15.02.10 Мехатроника и мобильная робототехника (по отраслям)».
13. Славинский, А.К. Электротехника с основами электроники: Учебное пособие / А.К. Славинский, И.С. Туревский - М.: ИД ФОРУМ, НИЦ ИНФРА-М, 2016. - 448 с.
14. Федотов, А.В. Основы теории автоматического управления. Учебное пособие / А.В. Федотов - Омск: Изд-во ОмГТУ, 2012. - 279 с.
15. Фельдштейн, Е.Э. Автоматизация производственных процессов в машиностроении: учеб. пособие / Е.Э. Фельдштейн, М.А. Корниевич. - Минск: Новое знание; М.: ИНФРА-М, 2017. - 264 с.
16. Шеховцов, В.П. Электрическое и электромеханическое оборудование: учебник / В.П. Шеховцов. - 3-е издание. - М.: ФОРУМ: ИНФРА-М, 2017. - 416 с.

17. Шишмарев, В.Ю. Автоматика. Учебник для среднего профессионального образования / В.Ю. Шишмарев. М.: Издательский центр «Академия», 2016. -288 с.